

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1.– 8. (Canceled)

19. (Currently amended) In a distributed computing system comprising a server cluster and a client computer, wherein the server cluster includes a first server and a second server, and wherein the client computer includes a client application, a cache, and a client-side network access module (NAM) by which the client application communicates with the server cluster, a method for providing seamless fail-over connectivity between the client computer and the server cluster, the method comprising:

receiving, at the client-side NAM, a request from the client application comprising a server name and a cluster name of the server cluster;

sending, by the client-side NAM, a first Server Resolution Protocol request in user datagram protocol (UDP) to the server cluster requesting first connection information for a server associated with the server name;

receiving, at the client-side NAM, a first Server Resolution Protocol reply comprising the connection information of the first server, wherein the connection information of the first server comprises a port number for a Virtual Interface Architecture (VIA) port of the first server;

establishing, by the client-side NAM, a first connection between the client application and the VIA port of the first server in the server cluster using a Virtual Interface Architecture (VIA) VIA protocol, wherein the VIA protocol assumes a single server is associated with the server cluster having a cluster name, and wherein the client-side NAM establishes the first connection using connection information of the first server in the server cluster to map the first server to a server name and the cluster name requested by the client application;

providing, by the client-side NAM, seamless fail-over connectivity from the first server to the second sever in the server cluster in a manner transparent to the client application, when the server cluster automatically switches operation from the first server to the second server, wherein providing seamless fail-over connectivity by the client-side NAM comprises:

detecting, by the client-side NAM, a failure of the first connection between the first server in the server cluster and the client application, when the server cluster automatically switches operation from the first server to the second server;

sending, by the client-side NAM, a second Server Resolution Protocol request in UDP to the server cluster requesting connection information for a new working server in the server cluster associated with the server name and the cluster name of the server cluster;

receiving, by the client-side NAM, a second Server Resolution Protocol ~~response~~ reply from the server cluster comprising connection information of the second server as the new working server, wherein the connection information of the second server comprises a port number for a VIA port of the second server;

caching, by the client-side NAM, the ~~received response~~ second Server Resolution Protocol reply in the cache of the client computer such that the cache contains a mapping between the connection information of the second server and the cluster name and the server name; and

establishing, by the client-side NAM, a second connection between the client application and the VIA port of the second server using the ~~Virtual Interface Architecture (VIA)~~ VIA protocol based on the connection information of the second server, wherein the server cluster does not provide fail-over support to re-direct a request from the client application from the first server to the second server when the server cluster automatically switches operation from the first server to the second server.

20. (Canceled)

21. (Previously Presented) The method of claim 19, wherein the establishing the second connection comprises retrieving a network address from the connection information of the second server from the cache.

22. (Previously Presented) The method of claim 21, further comprising verifying that the second server maps to the server name.

23. (Previously Presented) The method of claim 22, wherein verifying that the second server maps to the server name comprises verifying that the second server is responsive to the server name.

24. (Currently amended) The method of claim 19, further comprising purging the cache prior to caching the second Server Resolution Protocol reply ~~response from the second server~~.

25. (Currently amended) The method of claim 19, wherein the ~~Server Resolution Protocol request is sent by User Datagram Protocol (UDP)~~ the connection information provided in the first Server Resolution Protocol response comprises a plurality of port numbers, wherein each port number corresponds to a different communications protocol.

26. (Previously Presented) The method of claim 19, wherein the first server and second server store and retrieve relational data by way of Structured Query Language (SQL) commands.

27. (Previously Presented) The method of claim 19, wherein the first server is designated as active and the second server is designated as passive.

28. (Previously Presented) The method of claim 27, wherein the second server periodically sends a keep-alive message to the first server.

29. (Currently amended) The method of claim 28, wherein the second server assumes designation as active when the first server fails to ~~response~~ respond to the keep-alive message.

30. (Currently amended) The method of claim 19, wherein the connection information provided in the second Server Resolution Protocol response ~~contains~~ comprises a plurality of port numbers, wherein each port number ~~corresponded~~ corresponds to a different communications protocol.

31. (Previously Presented) The method of claim 19, wherein the first connection and the second connection comprise VIA formatted packets.

32. (Currently amended) A computer readable storage medium comprising a tangible physical structure having computer executable instructions that are executable by a computer to perform a method for providing seamless fail-over connectivity in a distributed system comprising a client computer and a server cluster, wherein the server cluster includes a first server and a second server, and wherein the client computer includes a client application, a cache, and a client-side network access module (NAM) by which the client application communicates with the server cluster, the method comprising:

receiving, at the client-side NAM, a request from the client application comprising a server name and a cluster name of the server cluster;

sending, by the client-side NAM, a first Server Resolution Protocol request in user datagram protocol (UDP) to the server cluster requesting first connection information for a server associated with the server name;

receiving, at the client-side NAM, a first Server Resolution Protocol reply comprising connection information of the first server, wherein the connection information of the first server comprises a port number for a Virtual Interface Architecture (VIA) port of the first server;

establishing, by the client-side NAM, a first connection between the client application and the VIA port of the first server in the server cluster using a Virtual Interface Architecture (VIA) VIA protocol, wherein the VIA protocol assumes a single server is associated with the server cluster having a cluster name, and wherein the client-side NAM establishes the first

~~connection using connection information of the first server in the server cluster to map the first server to a server name and the cluster name requested by the client application;~~

providing, by the client-side NAM, seamless fail-over connectivity from the first server to the second sever in the server cluster in a manner transparent to the client application, when the server cluster automatically switches operation from the first server to the second server, wherein providing seamless fail-over connectivity by the client-side NAM comprises:

detecting, by the client-side NAM, a failure of the first connection between the first server in the server cluster and the client application, when the server cluster automatically switches operation from the first server to the second server;

sending, by the client-side NAM, a second Server Resolution Protocol request in UDP to the server cluster requesting connection information for a new working server in the server cluster associated with the server name and the cluster name of the server cluster;

receiving, by the client-side NAM, a second Server Resolution Protocol ~~response~~ reply from the server cluster comprising connection information of the second server as the new working server, wherein the connection information of the second server comprises a port number for a VIA port of the second server;

caching, by the client-side NAM, the ~~received response~~ second Server Resolution Protocol reply in the cache of the client computer such that the cache contains a mapping between the connection information of the second server and the cluster name and the server name; and

establishing, by the client-side NAM, a second connection between the client application and the VIA port of the second server using the ~~Virtual Interface Architecture (VIA)~~ VIA protocol based on the connection information of the second server, wherein the server cluster does not provide fail-over support to re-direct a request from the client application from the first server to the second server when the server cluster automatically switches operation from the first server to the second server.

33. (Canceled)

34. (Previously presented) The computer readable storage medium of claim 32, wherein the first connection and the second connection comprise VIA formatted packets.

35. (Currently amended) A system for providing seamless fail-over of client connectivity with a server cluster, the system comprising:

a client application;

a client-side network access module (NAM) by which the client application communicates with the server cluster; and

a cache,

wherein the client-side network access module;

receives a request from the client application comprising a server name and a cluster name of the server cluster,

sends a first Server Resolution Protocol request in user datagram protocol (UDP) to the server cluster requesting connection information for a server associated with the server name,

receives a first Server Resolution Protocol reply comprising connection information of a first server, wherein the connection information of the first server comprises a port number for a Virtual Interface Architecture (VIA) port of the first server,

establishes a first VIA protocol connection between the client application and the VIA port of the first server in the server cluster using a VIA protocol,

~~(i) detects a failure of [[a]] the first Virtual Interface Architecture (VIA) protocol connection between the client application and [[a]] the first server in the server cluster, wherein the first connection is established using connection information of the first server which is mapped to a server name and cluster name of the server cluster;~~

(ii) sends a second Server Resolution Protocol request in UDP to the server cluster requesting connection information for a server associated with the server name and the cluster name of the server cluster,

(iii) receives a second Server Resolution Protocol ~~response~~ reply from the server cluster comprising connection information of a second server, wherein the connection information of the second server comprises a port number for a VIA port of the second server,

(iv) caches the ~~received response~~ second Server Resolution Protocol reply in the cache such that the cache contains a mapping between the connection information of the second server and the cluster name and the server name, and

(v) establishes a second ~~Virtual Interface Architecture~~ VIA protocol connection between the client application and the second server using the connection information of the second server,

wherein the server cluster does not provide fail-over support to re-direct a request from the client application from the first server to the second server when the server cluster automatically switches operation from the first server to the second server; and

wherein the detecting, the sending, the receiving, caching, and the establishing are performed by the client-side NAM to provide seamless fail-over connectivity of the client application from the first server to the second sever in a manner transparent to the client application, when the server cluster automatically switches operation from the first server to the second server.

36. (Canceled)

37. (Previously Presented) The system of claim 35, wherein the first connection and the second connection comprise VIA formatted packets.